**Gen Z 세대를 위한 공유 캘린더**

[간신히 Gen z]  
요구사항 분석서

2024-10-01

2271246 김소진, 2071111 이준용, 2091019 채희석, 2071012 이연준

## 프로젝트 목표 정의

사용자들로 하여금 캘린더 즉 달력을 활용하여 일정관리를 한눈에 할 수 있도록 하고, 그룹을 형성하여 이 그룹내에 서로 일정을 공유하는 공유 캘린더의 역할, 그리고 공유그룹에 포함된 인원들의 일정을 자동으로 종합해 최적의 날짜, 최적의 시간대에 모임시간 및 약속 일정을 잡아주는 기능을 추가한 다목적용 공유 캘린더를 만드는 것이다.

## 요구사항 수집

**2.1 인터뷰 질문 리스트**

현재 확정적으로 추가될 기능은 다음과 같습니다.

1. 회원가입 및 로그인 기능

2. 일정 추가 및 삭제 기능

3. 공유 그룹 생성 및 초대, 삭제 기능

4. 구글 캘린더와의 연동

5. 캘린더 일정 공개, 비공개 선택 기능

6. 같은 그룹 내의 약속 자동 생성 기능

7. 한눈에 보이는 내 일정 화면

공유 캘린더 형식을 띄고 있고 서로의 일정을 확인할 수 있고, 서로의 일정을 바탕으로 자동적으로 약속을 잡아줄 수 있도록 해주는 기능을 포함하고 있습니다.

UI가 어떤 형식이었으면 좋겠는가?

위의 기능을 제외한 어떤 기능이 추가되면 좋겠는가?

위의 기능 중 어떤 기능이 가장 중요하게 느껴지는가?

**2.2 인터뷰 내용**

공유 그룹 생성에 있어서 어떤 방식으로 초대할 것인지가 중요해 보이고 초대 및 수정이 편리하게 이루어지면 좋겠다.

공유 캘린더인 만큼 일정 변경이 실시간으로 잘 이루어져야 하고, 직관적으로 일정을 확인할 수 있어야 한다.

약속 자동 생성기능에 있어서 만약 불가피하게 한 명이 급하게 일정을 바꾸게 된다면 어떻게 일정을 새로 잡을지 중요해 보인다.

그리고 전체적으로 그룹생성, 약속자동생성 등 어떠한 기능이 이루어졌을 때 알림을 통하여 확실히 전달할 필요가 있어 보인다는 반응이었다.

**2.3 프로젝트 용어사전**

**DB(Data Base)**: 여러 사람에 의해 공유되어 사용될 목적으로 통합하여 관리되는 데이터의 집합을 말한다.

**UI(User Interface)**: 사용자 인터페이스로서 사용자와 컴퓨터 시스템 사이의 의사소통 매개, 즉 해당 기기에 대해 효율적인 작동과 사용자의 제어를 보장한다.

**UX(User experience)**: 사용자 경험으로써 사용자가 어떤 시스템, 서비스 및 제품 등을 직, 간접적으로 이용하면서 느끼는 총체적 경험이다.

**데이터 무결성**: 컴퓨팅 분야에서 완전한 수명 주기를 거치며 데이터의 정확성과 일관성을 유지하고 보증하는 것이다

**API(Application Programming Interface):** 정의 및 프로토콜 집합을 사용하여 두 소프트웨어 구성 요소가 서로 통신할 수 있게 하는 메커니즘이다.

**RF(Requirement Functionality):** 요구사항을 체계적으로 관리하고 추적할 수 있도록 기능 요구사항의 약어로 번호를 매겨 요구사항을 구분하기 위해서 사용한다.

## 요구사항 분석

**3.1 요구사항 정렬**

RF 1.0 캘린더 기능

-- RF 1.1 캘린더 보기

-- RF 1.2 일정 등록

-- RF 1.3 일정 삭제

-- RF 1.4 일정 변경

-- RF 1.5 한눈에 캘린더 보기

-- RF 1.6 할 일, 루틴, 습관 등록

-- RF 1.7 일정 메모

RF 2.0 사용자 등록 기능

-- RF 2.1 회원가입

-- RF 2.2 로그인

-- RF 2.3 사용자 정보 저장

RF 3.0 공유 그룹 기능

-- RF 3.1 그룹 생성

-- RF 3.2 그룹 삭제

-- RF 3.3 그룹 초대 및 참여

-- RF 3.4 그룹 탈퇴 및 제외

-- RF 3.5 그룹 내 일정 조율

-- RF 3.6 그룹 참여자 확인

RF 4.0 알림 기능

-- RF 4.1 일정 알림 기능

RF 5.0 일정 관리 기능

-- RF 5.1 일정 카테고리 분류

-- RF 5.2 일정 분석

-- RF 5.3 일정 공개여부 설정

RF 6.0 동기화 및 연동 기능

-- RF 6.1 캘린더 동기화

-- RF 6.2 캘린더 연동

RF 7.0 비기능적 요구사항

-- RF 7.1 응답 속도 향상

-- RF 7.2 보안 사항

-- RF 7.3 호환성

-- RF 7.4 사용자 친화성

-- RF 7.5 안정성 및 확장성

-- 요구사항 정의서에서 자세히 기술하였음 --

## 요구사항 구분

**4.1 기능적 요구사항**

--할 일, 루틴, 약속 분류: 일정 등록 시 할 일, 약속으로 분류할 수 있어야 한다.

--일정 추가: 터치 앤 드래그로 쉽게 일정을 추가할 수 있어야 한다.

--일정 자동 조율: 그룹 내 사용자의 일정을 자동으로 조율하고, 충돌 발생 시 다른 일정 후보를 제시하며 자동으로 일정을 재조정해야 한다.

--알림: 사용자는 중요한 일정, 약속 변경 사항 등을 적시에 알림을 통해 받을 수 있어야 하며, 알림 방식(이메일, 푸시 알림 등)을 선택할 수 있어야 한다.

--회원가입 및 로그인: 사용자 계정 관리 기능이 필요하다.

--공유 그룹 기능: 그룹을 생성하고, 그룹 내 일정 고유 및 초대/삭제 기능이 필요하다.

**4.2 비기능적 요구사항**

--응답 속도: 일정 조율 및 일정 추가 시 빠른 응답 시간이 필요하다.

--보안: 일정 및 개인 정보 보호를 위한 암호화 필요하다.

--호환성: 다양한 운영체제 및 기기에서 호환 가능해야 한다.

--안정성과 데이터 무결성: 다수의 일정 등록 및 수정 시 데이터 무결성을 보장해야 하며, 시스템은 이러한 작업 중 안정적으로 동작해야 한다.

--확장성: 더 많은 사용자 및 그룹을 지원할 수 있도록 확장 가능해야 한다.

--디자인: 사용자와 공유자의 일정을 쉽게 확인할 수 있도록 심플하고 직관적이어야 한다.

**4.3 시스템 및 사용자의 요구사항으로 분류**

**4.3.1 시스템 요구사항**

--데이터 베이스: 사용자 정보, 일정 그룹 정보를 효율적으로 저장하고 관리할 수 있어야 한다.

--API 연동: 구글 캘린더와 같은 외부 캘린더 시스템과의 연동이 가능해야 한다.

--알림 시스템: 푸시 알림 기능을 통해 중요한 일정 및 일정 변경 사항을 알릴 수 있어야 한다.

--보안: 일정 및 개인정보 보호를 위한 강력한 보안 시스템을 구축해야 한다.

**4.3.2 사용자 요구사항**

--사용자 경험: 사용자가 쉽게 자신의 일정을 추가하고 공유할 수 있어야 한다.

--그룹 관리: 사용자는 그룹을 쉽게 생성하고 관리할 수 있어야 한다,

--일정 조율: 여러 사용자의 일정이 자동으로 조율될 수 있어야 한다.

## 프로젝트 상세 기술 사항

**5.1 프로젝트 어플리케이션 관점**

공유 캘린더 어플리케이션은 기존에 있는 캘린더를 넘어서 공유그룹을 형성하여 공유 그룸 내의 인원들의 공유 일정을 잡아주는 기능을 핵심으로 생각하고 있다. 보다 더 나은 사용자 편의성과 다른 외부 캘린더에서 제공하는 좋은 기능들과 더불어 추가적인 기능을 추가해 사용자의 편의성을 모두 챙기는데 의의를 두고있다.

**5.2 프로젝트 어플리케이션 구조**

도표, 텍스트, 평면도, 기술 도면이(가) 표시된 사진

자동 생성된 설명

**그림 1. 어플리케이션 구조 다이어그램**

-- 데이터 흐름도는 다음장인 데이터 흐름도 참고 –

**5.3 프로젝트 데이터 흐름도**

**5.4 사용자 특성**

어플리케이션 사용자는 어플을 통해 본인의 일정을 마음대로 추가, 변경, 삭제 할 수 있어야 한다. 또한 공유그룹을 자유롭게 생성하고 삭제 할 수 있어야 하며, 공유 일정을 생성하였을때 성공적으로 사용자 개인 캘린더에 반영될 수 있어야 한다. 모든 기능이 간편하고 직관적이여야 하며, 사용자 친화성이 높아야 한다.

**5.5 제약사항**

- 안드로이드 스튜디오를 이용해 안드로이드 기반 기기에서 구현할 수 있도록 한다.

- 사용자 데이터를 관리할 데이터 베이스는 MySQL을 기반으로 한다.

- 서버 구현 시 Java를 이용한 Spring boot를 통해 구현한다.

- 외부 캘린더와의 연동은 연동이 비교적 쉬운 Google Calendar로 한정한다.

- 공유 일정 선정 알고리즘은 Open API를 사용하거나, 직접 코딩하여 구현한다.

**5.6 가정 및 의존성**

사용자는 본인의 일정을 모두 캘린더에 작성하는것을 가정한다. 기입되지 않거나 등록되지 않은 일정은 고려하지 않는다.

공유 그룹 생성에 대해 사전에 협의가 된 그룹이 형성된다는 것을 전재로 하여, 무분별한 그룹 초대 및 원하지 않는 그룹 참여를 제한하고, 공개하고싶지 않은 사람에게 사용자의 일정을 공유하지 않는다는 것을 가정한다.

안드로이드 기반 기기로 구현하기 때문에 안드로이드 시스템에만 기반되는 의존성이 존재한다.

(개발 추후 멀티 플랫폼에 관해서 고려해볼 수 있음)

## 상세 기능 요구사항

**6.1 요구사항 정의서**

**6.1.1 RF 1.0 캘린더 기능**

|  |  |
| --- | --- |
| **구분** | **내용** |
| **요구사항 NO** | RF 1.1 |
| **요구사항 명** | 캘린더 |
| **상세 요구사항 정의** | 캘린더(달력) 및 날짜, 시간을 볼 수 있어야한다. |
| **유형** |  |
| **구분** |  |
| **도출단계** |  |
| **상태** |  |
| **출처** |  |
| **제약사항** |  |
| **수용여부** |  |
| **변경 내역** |  |